



TITLE:

凸多角形の多角形領域内への
maximin配置問題とそれに関連し
た動的Voronoi図(アルゴリズムと
計算量の理論)

AUTHOR(S):

青沼, 裕美; 今井, 浩; 今井, 桂子; 徳山, 豪

CITATION:

青沼, 裕美 ...[et al]. 凸多角形の多角形領域内へのmaximin配置問題とそれに関連した動的Voronoi図(アルゴリズムと計算量の理論). 数理解析研究所講究録 1990, 731: 177-186

ISSUE DATE:

1990-10

URL:

<http://hdl.handle.net/2433/101968>

RIGHT:

凸多角形の多角形領域内への maximin 配置問題とそれに関連した動的 Voronoi 図

青沼裕美*, 今井浩*, 今井桂子**, 徳山豪***

*九州大学工学部情報工学科

**九州工業大学情報科学センター

***日本 IBM 東京基礎研究所

Maximin Polygon Containment Problems and Related Dynamic Voronoi Diagrams

Hiromi Aonuma*, Hiroshi Imai*, Keiko Imai**, Takeshi Tokuyama***

*Department of Computer Science and Communication Engineering, Kyushu University

**Information Science Center, Kyushu Institute of Technology

***IBM Research, Tokyo Research Laboratory

1. はじめに

多角形配置問題とは、与えられた多角形 P を他の多角形 Q の内部に配置するという問題である。この問題は、モーシヨンプランニングとも密接に関係があり、いろいろな面から研究がなされている ([F1] や [CK] 参照)。

本稿では、何種類かの多角形 maximin 配置問題を考える。最も基本的な maximin 配置問題は、 P の任意の点と Q の任意の点の Euclid 距離の最小値が最大となるように凸多角形 P を多角形 Q の内部に配置するという問題である。さらに、 P をいくつか一直線上に並ぶようにして、同様の maximin 配置問題を解くことも考える。直感的に言えば、多角形 P またはそのいくつかのコピーを Q の境界からなるべく離れるように Q の内部に置く問題を考えることであると言ってよい。似たような問題に、 P に相似な最大の多角形を Q の内部に配置するという問題があり、これについては Fortune [F1] や Chew と Kedem [CK] で考察されている。しかし、彼らは凸距離関数を用いており、Euclid 距離を用いた問題については、これまでに研究がなされていない。

ここでは、新しい Voronoi 図 (P-Euclid Voronoi 図と呼ぶ) を定義することによってこれらの maximin 配置問題に対する効率の良いアルゴリズムを紹介し、それらの組合せ的複雑度についての解析を行なう。計算複雑度の解析においては、Davenport-Shinzel 列の理論を用いる。また、 n 個の点からなる k 個の集合が動く場合の通常の意味での Voronoi 図の組合せ的複雑度についても考察した。

凸多角形の多角形内への配置問題は、地図の中へ地名などを記入するときにも現われる。これは、地図データベースにおいて、縮尺や用途に応じて図形と文字を選択し合成するという利用者インタフェースを実現する上での重要な問題である ([AIK])。このような場面では、文字列は長方形で表わされ、領域は多角形で表現される。長方形を、上記の意味での

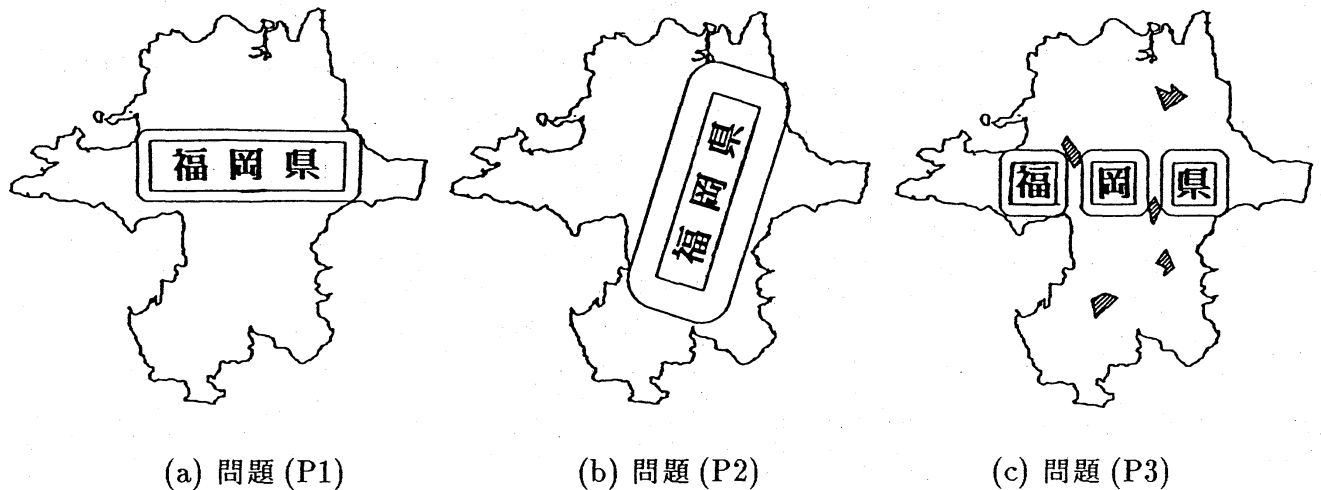


図 1. 地図に地名を配置する問題

maximin 問題の解となるように多角形の内部に置きたい. また, 次のような配置問題もある. 文字列を 1 つの長方形として捕らえるのではなく, すでに置かれている文字と重ならないように, かつ, 一定の間隔で一列に並ぶように各文字を置くと考えるのである. このような問題を解くためには, 穴のある多角形の内部に正方形のコピーを一定の間隔で配置する問題を考える必要がある.

本稿で扱う maximin 配置問題と得られた結果, また, それに関連してすでに得られている結果をまとめると次のようになる. P は m 個の辺を持つ凸多角形, Q は n 個の辺を持つ多角形 (または多角形領域) とする.

(P1) 平行移動だけを用いて P の任意の点と Q の任意の点との Euclid 距離の最小値が最大となるように凸多角形 P を多角形領域 Q の内部に置く.

問題 (P1) は, 平行移動により P の相似な図形のうち最大のものを Q の内部に配置する問題と関係が深い. P の相似な図形で最大のものを配置する問題については P に関する凸距離関数に対する Q の Voronoi 図を用いて $O(mn \log mn)$ の手間で解ける ([F1]). 本稿では, 問題 (P1) に対して新しい Voronoi 図を定義して, この Voronoi 図が [F1] の Voronoi 図を求めるアルゴリズムと, $O(n)$ 本の線分に対する Euclid 距離の Voronoi 図を構成する $O(n \log n)$ の手間のアルゴリズム ([F2] 参照) を組み合わせることによって, $O(mn \log mn)$ の手間で構成することができることを示す. この新しい Voronoi 図が与えられれば, (P1) は線形時間の手間で解けるので, 結局 (P1) は全体で $O(mn \log mn)$ の手間で解ける.

(P2) 回転と平行移動によって P の任意の点と Q の任意の点との Euclid 距離の最小値が最大になるように凸多角形 P を多角形領域 Q 内に配置せよ.

Chew と Kedem [CK] では, 回転と平行移動によって P と相似な多角形で最大のものを Q の中に置くという (P2) に類似した問題を扱っている. しかし, 凸多角形 P が細長い場合には, (P2) のように P を置く方が自然である. (P1) で Voronoi 図の動的な場合を考えることによって問題 (P2) を $O(m^4 n \lambda_{16}(mn) \log mn)$ の手間で解くことができる.

ここで、 $\lambda_{16}(mn)$ は位数 16 の Davenport-Schinzel 列の最大長を表わす。一般に、 $\lambda_s(N)$ は N にほとんど線形な関数であることがわかっている。

(P3) Q 内に P の k 個のコピーを次のように配置する。 P のコピーは水平線上に並び、コピー間の距離 h と、 P の任意の点と Q の任意の点の間の距離の最小値が最大となるようにする。ここで、 h は与えられた定数 $h_0 > 0$ 以上であるような変数である。

h を固定した場合は、この問題は多角形領域 Q の k 個のコピーを h ずつずらして置いた時の交わりを計算し、この交わりに対して (P1) を考えることによって解くことができる。 h が可変の場合のこの問題では、 Q の k 個のコピーを間隔 h で置いたときの Voronoi 図を考え、それを h を h_0 から ∞ へ動かしたときの動的な Voronoi 図について考察し、この Voronoi 図の組合せ的複雑度を評価する。 $k=2$ と $k \geq 3$ の場合にそれぞれ問題 (P3) は、 $O(m^2 n^2 \log mn)$ 、 $O(k^6 m^3 n^3 \log kmn)$ の手間で解ける。

この動的 Voronoi 図の組合せ的複雑度の解析において、この種の問題のうち最も基本的であると思われる次の問題を解き、そのアルゴリズムを拡張して、問題 (P3) を解いた。

(P4) n 個の点から成る k 個の集合を考える。各集合内では点の相対的な位置は変わらず、 k 個の集合はその性質を保ったまま次数が n に無関係な h の低次の多項式や h の連続な有理関数によって動く。このとき、動的な Voronoi 図の組合せ的複雑度を評価せよ。

これらの関数が良い性質を持ち、それらに関する方程式が定数時間で解くことができると仮定すれば、この Voronoi 図の組合せ複雑度は、 $k=2, 3$ に対して $O(n^2)$ 、 $k \geq 4$ の時は $O(k^4 n^3 \log^* k)$ である。

本稿では、新しい Voronoi 図の定義を行ない、それをもとに問題 (P1), (P2) を解く方法についての概説を述べ、(P3), (P4) については結果を述べるにとどめる。

2. P-Euclid Voronoi 図について

この節では、問題 (P1), (P2), (P3) を解くために新しい Voronoi 図 (P-Euclid Voronoi 図) を定義する。そのために、まず、 Q 内の P の配置可能領域について述べる。凸多角形 P は、 P の内部に参照点 p を持つと仮定する。平面上の点 q に対して、多角形 P の参照点 p が q の位置に来るように P を平行移動したものを $P(q)$ と書くことにする。 P の Q 内の配置可能領域とは、 $P(q)$ が Q に含まれるような点 $q \in Q$ の集合のことである。配置可能領域は、境界が多角形であるようないくつかの連結成分から成る。配置可能領域の境界を $F(P, Q)$ で表わすことにする。配置可能領域の組合せ的複雑度 (ここでは $F(P, Q)$ の辺と頂点の数) は $O(mn)$ であり、 P に関する凸距離関数による Q の Voronoi 図を利用して、 $F(P, Q)$ を $O(mn \log mn)$ の手間で構成することができる ([F1] 参照)。

点 p_f が $F(P, Q)$ の辺 e_f の上にあるとすれば、多角形 $P(p_f)$ は Q 内にあり、 $P(p_f)$ の頂点または辺が Q の頂点か辺に接している。もし、 $P(p_f)$ の頂点と Q の頂点が接しているとすれば、 p_f は $F(P, Q)$ の頂点である。 P の頂点と Q の辺が接しているとすれば、

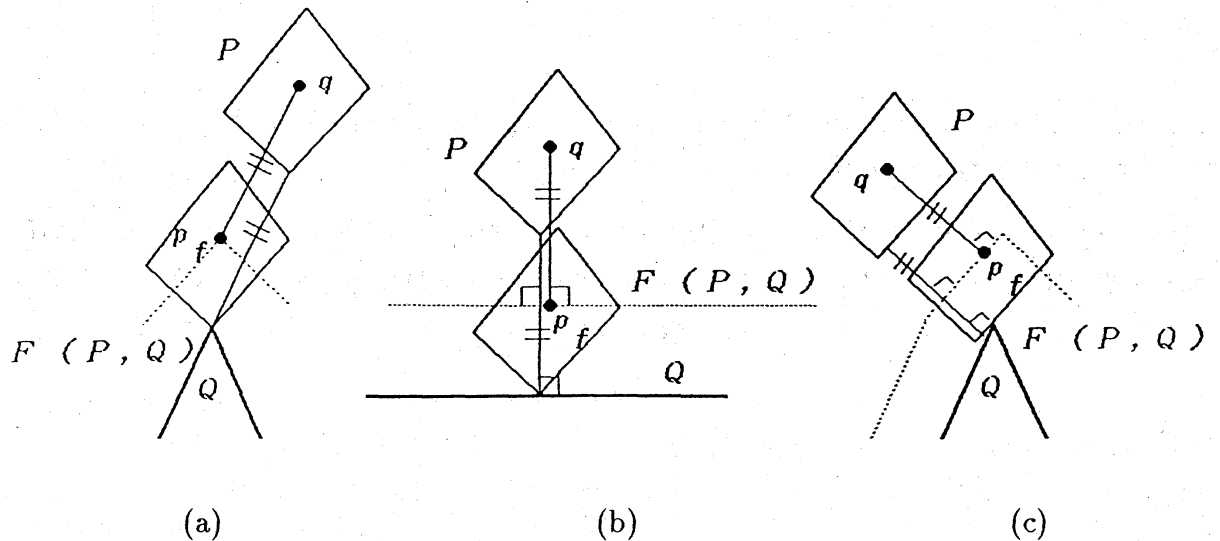


図 2. 補題 2.1 の証明

e_f は Q の辺に平行である. P の辺と Q の凹頂点が接しているときは, e_f は P の辺に平行である.

点 $q \in F(P, Q)$ に対して, Q の点と $P(q)$ の点の間の Euclid 距離の最小値を $d_P(Q, q)$ で表す. また $d(F, q)$ は $F(P, Q)$ の点と q の間の Euclid 距離の最小値を表わすとする. このとき, 配置可能領域は次の性質を持つ.

補題 2.1. 配置可能領域内の点 q に対し, $d_P(Q, q) = d(F, q)$ が成り立つ.

配置可能領域 $F(P, Q)$ に対し, Euclid Voronoi 図を考え, これを P-Euclid Voronoi 図と呼ぶことにする. この P-Euclid Voronoi 図においては, $F(P, Q)$ の辺は開線分と 2 つの頂点から構成されたと考え, Voronoi 領域は, それらの開線分と頂点に対する領域から成る. P-Euclid Voronoi 図は $O(mn)$ の Voronoi 領域と辺と頂点からできている.

Q の辺 I_i ($i = 1, \dots, n$) に対し, $d_P(I_i, q) \leq d_P(I_j, q)$ ($j \neq i$) となるような $q \in F(P, Q)$ の集合 $V(I_i)$ を考える. P-Euclid Voronoi 図の領域が, 辺 e の領域とすると, e 上に参照点をおいた時, P が接する Q の辺 I_i に対し, 補題 2.1. より, e の Voronoi 領域は $V(I_i)$ に一致する. 従って, P-Euclid Voronoi 図とは, Q の辺 I_i (または頂点) に対して, 辺 I_i の任意の点と $P(q)$ の任意の点との距離 (または Q の頂点と $P(q)$ の任意の点との距離) が, Q の他の辺や点に対する距離より小さいような q の集合を Q の辺 I_i (または頂点) の Voronoi 領域とする Voronoi 図と同じである.

3. 平行移動を用いた maximin 配置問題

問題 (P1) に対しては次の定理が成り立つ.

定理 3.1. 問題 (P1) は, P-Euclidean Voronoi 図を用いることにより $O(mn \log mn)$ の手間で解くことができる.

補題 2.1 により, 問題 (P1) は $F(P, Q)$ の任意の点との距離の最小値が最大となるような点 $q \in F(P, Q)$ を見つけることに帰着できる. これは, $F(P, Q)$ に含まれる最大円を見

つけることに他ならない. 多角形 $F(P, Q)$ に対する P-Euclid Voronoi 図は $O(mn \log mn)$ で構成できる ([F2]). そして, P-Euclid Voronoi 図が与えられれば, $F(P, Q)$ に含まれる最大円は線形時間で求められる.

4. 平行移動と回転を用いた maximin 配置問題

問題 (P1) においては, P-Euclid Voronoi 図を求めることにより, 問題を解くことができた. しかし, P の向きが変わるとき, 配置可能領域それ自身が変化し, P-Euclid Voronoi 図は動的に変わってゆく. 問題 (P2) をこの P-Euclid Voronoi 図を使って解くために, この動的 P-Euclid Voronoi 図について調べなければならない. Chew と Kedem [CK] によって解かれている Q 内の P と相似であり大きさが最大のものを見つける問題に置いても同じようなことが問題となっている. そこで, [CK] で用いられたアプローチを基にして, 動的な P-Euclid Voronoi 図の組合せ的複雑度の解析を行なうことにする.

P を参照点 p を中心として θ ($0 \leq \theta \leq 2\pi$) 回転したものを $\tilde{P}(\theta)$ と書くことにする. 配置可能領域 $F(\tilde{P}(\theta), Q)$ の点 q に対して, 参照点 p が q の位置に来るように $\tilde{P}(\theta)$ を平行移動したものを $\tilde{P}(\theta, q)$ で表わす.

θ が変わるとき $\tilde{P}(\theta)$ の配置可能領域 $F(\tilde{P}(\theta), Q)$ も変化する. この節では, まず, 配置可能領域の変化について調べることにする.

Voronoi 辺上に $\tilde{P}(\theta)$ の参照点を置くと, $\tilde{P}(\theta, q)$ の任意の点と Q の任意の点と距離の最小値をとる点の組は, 2 組ある. それらの点は, P , Q の頂点であるかまたは P , Q の辺上にある. このときの P の頂点 (辺) を r , r' , Q の頂点 (辺) を s , s' とした時, (r, s) と (r', s') は, その Voronoi 辺を支持するということにする. r と r' , s と s' は一致するかもしれない. $r = r'$ となる時は r は P の頂点であり, $s = s'$ となる時は s は Q の凹頂点となっている. そこで, Voronoi 辺を支持する組が, $\tilde{P}(\theta, q)$ の頂点 p^* と Q の 2 つの頂点 (辺) である時, または, $\tilde{P}(\theta, q)$ の 2 つの頂点 (辺) と Q の凹頂点である時, q を通っている Voronoi 辺を b-edge ということにする.

配置可能領域 $F(\tilde{P}(\theta), Q)$ の組合せ的複雑度は $O(mn)$ であるので, そのことから, 次の補題が得られる.

補題 4.1. b-edge の数は, 動的 Voronoi 図において $O(mn)$ である.

次に, 動的 Voronoi 図全体をとおして, 次数が 4 以上になる Voronoi 点はいくつあるかを数える. Voronoi 点の次数が 4 以上になるとは, 隣接する 2 つの Voronoi 点が一一致して, その間の Voronoi 辺が長さ 0 になり退化することを意味する. この回数が $\tilde{P}(\theta)$ -Euclid Voronoi 図の組合せ的複雑度の上界を与える. 次数が 4 以上である Voronoi 点を v-point と呼ぶ.

複雑度の評価は, 次の 2 つのステップで行なう.

(step 1) ある b-edge に対して, この b-edge の端点 v-point となる回数は, $O(\lambda_{16}(mn))$ である. 従って, b-edge の端点 v-point となる回数の総数は, $O(mn\lambda_{16}(mn))$ である.

(step 2) b-edge の端点になっていない v-point の数は, step 1 の総数に m^3 を掛けた数でおさえられる.

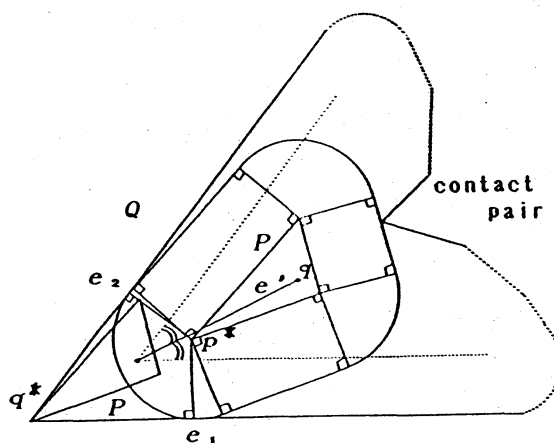


図 3. タイプ (1) の b-edge e' と対応する P の頂点 p^*

(step 1)

step 1 では b-edge の端点となっている v-point の数を数える. b-edge には, 次の 4 つのタイプがある.

- (1) P の頂点と Q の 2 つの辺で支持される b-edge
- (2) P の頂点とそれに隣接する P の辺と Q の凹頂点で支持される b-edge, または, P の頂点と Q の凹頂点とそれに隣接する Q の辺で支持される b-edge
- (3) P の頂点と Q の凹頂点とそれには隣接していない Q の辺で支持される b-edge
- (4) P の頂点と Q の 2 つの凹頂点で支持される b-edge

Voronoi 辺 e' がタイプ (1) の b-edge であるとする. Q の 2 つの支持辺を e_1 と e_2 と書くことにする (図 3 参照). P の頂点か辺と Q の頂点か辺の組 c に対し, $p^* \in \tilde{P}(\theta, q)$ と e_1 との距離, $p^* \in \tilde{P}(\theta, q)$ と e_2 との距離, それに $\tilde{P}(\theta, q)$ と q の接する組 c の間の距離が等しくなるように $q \in F(\tilde{P}(\theta), Q)$ を取ったとき, その距離 δ の最小値で関数 $d_c(\theta)$ を定義する. 図 3 は, 接している組が, $\tilde{P}(\theta, q)$ の辺と Q の凹頂点である場合の例である. C は, P の頂点または辺と Q の頂点または辺のすべての組の集合を表わすとし, 次のような関数を定義する.

$$d_C(\theta) = \min_{c \in C} d_c(\theta)$$

もし, 接している組 c に対する d_c が d_C の最小値を取るとすると, この c と p^* , e_1 と e_2 は b-edge の Voronoi 点を定める. θ が変わるにつれ, 最小値も変化する. d_C は Davenport-Schinzel 列を用いて, $d_c(\theta)$ ($c \in C$) の下側エンベロップから求める. $d_c(\theta)$ は, 接している組の組合せにしたがって一般に次のような形をしている.(添字付きのギリシャ文字は定数を表わしている.)

(a) c が P の頂点と Q の辺の組であるとき: $d_c(\theta) = \alpha_1 \cos(\theta + \alpha_2) + \alpha_3$

(b) c が P の辺と Q の頂点の組であるとき:

$$d_c(\theta) = \frac{\beta_1 \cos(\theta + \beta_2) + \beta_3}{\beta_4 \cos(\theta + \beta_5) + \beta_6}$$

(c) c が P の頂点と Q の頂点の組であるとき :

$$(d_c(\theta))^2 + (\gamma_1 \cos(\theta + \gamma_2) + \gamma_3)d_c(\theta) + \gamma_4 \cos(\theta + \gamma_5) + \gamma_6 = 0$$

これらの mn 個の関数 $d_c(\theta)$ ($c \in C$) は互いに高々4回しか交わらない. Davenport-Shinzel 列に関する理論から, 次の補題が得られる.

補題 4.2. タイプ (1) の b-edge に対して, d_C の組合せ的複雑度は $O(\lambda_4(mn))$ である.

タイプ (2) の b-edge に対して, $d_c(\theta)$ を求めると, (a), (b) の場合には, タイプ (1) と同様な式となり, (c) の場合には

$$d_c(\theta) = \frac{\gamma_1 \cos(\theta + \gamma_2) + \gamma_3}{\gamma_4 \cos(\theta + \gamma_5) + \gamma_6}$$

と表わすことができる. これらの関数も互いに高々4点でしか交わらない. よって, 次の補題が得られる.

補題 4.3. タイプ (2) の b-edge に対して, d_C の組合せ的複雑度は $O(\lambda_4(mn))$ である.

タイプ (3) の b-edge に対して, $d_c(\theta)$ を求めると次の式で表わすことができる. ここで, 変数 x を次のようにおき, x で $d_c(\theta)$ を表わすことにする.

$$d_c(\theta) = x^2 + 1 \quad (x \geq 0)$$

(a) c が P の頂点と Q の辺の組であるとき :

$$x^2 + \alpha_1 x + \alpha_2 \cos(\theta + \alpha_3) + \alpha_4 = 0$$

(b) c が P の辺と Q の頂点の組であるとき :

$$(\cos(\theta + \beta_1) + \beta_2)x^2 + \beta_3 x \sin(\theta + \beta_1) + \beta_4 \cos(\theta + \beta_5) + \beta_6 = 0$$

(c) c が P の頂点と Q の頂点の組であるとき :

$$(\cos(\theta + \gamma_1) + \gamma_2)x^2 + (\gamma_3 \cos(\theta + \gamma_4) + \gamma_5)x + \gamma_6 \cos(\theta + \gamma_7) + \gamma_8 = 0$$

これらの関数は互いに高々8点でしか交わらない. よって, 次の補題が得られる.

補題 4.4. タイプ (3) の b-edge に対して, d_C の組合せ的複雑度は $O(\lambda_8(mn))$ である.

最後に, タイプ (4) の b-edge に対して, $d_c(\theta)$ を求める. ここでは, 変数 x を次のようにおき, x で $d_c(\theta)$ を表わす.

$$d_c(\theta) = \sqrt{x^2 + 1} \quad (x \geq 0)$$

(a) c が P の頂点と Q の辺の組であるとき :

$$x^2 + (\alpha_1 \cos(\theta + \alpha_2) + \alpha_3)x + \alpha_4 \cos^2(\theta + \alpha_2) + \alpha_5 \cos(\theta + \alpha_2) + \alpha_6 = 0$$

(b) c が P の辺と Q の頂点の組であるとき :

$$x^2 \sin^2(\theta + \beta_1) + \cos(\theta + \beta_1)(\beta_2 \cos(\theta + \beta_3) + \beta_4)x + \beta_5 \cos^2(\theta + \beta_3) + \beta_6 \cos(\theta + \beta_3) + \beta_7 = 0$$

(c) c が P の頂点と Q の頂点の組であるとき :

$$(\cos(\theta + \gamma_1) + \gamma_2)x + \gamma_3 \cos(\theta + \gamma_4) + \gamma_5 = 0$$

これらの関数は互いに高々16点で交わる．よって，次の補題が得られる．

補題 4.5. タイプ (4) の b-edge に対して, d_C の組合せ的複雑度は $O(\lambda_{16}(mn))$ である．

補題 4.2, 4.3, 4.4 と 4.5 をまとめると次の補題となる．

補題 4.6. 1 つの b-edge の端点で v-point となる回数は $O(\lambda_{16}(mn))$ である．

(step 2)

step 2 では, b-edge の端点となっていない v-point について調べる．まず, θ を固定し, すべての b-edge とそれに隣接する Voronoi 辺を取り除き, 残った Voronoi 辺を連結成分に分解する．残っている Voronoi 辺上に P の参照点を置くと, P から等距離にある Q の辺または頂点が丁度 2 つ存在する．いま考えている Voronoi 辺は b-edge ではないので, この 2 つの辺や頂点は P の同じ頂点には接していない．2 つの辺か頂点に接した状態を保ったまま, 参照点を Voronoi 辺上を動かしていく．すると, P の辺か頂点が Q の辺か頂点に接して P は止まる．それ以上最初の 2 つの辺または頂点に接した状態のまま P を動かすことはできない．そこで, Voronoi 点で P が停止したら, 接している辺や頂点の組を取り替えて, 再び P を動かす．支持している辺, 頂点が b-edge に対応するものになると, その先には進めなくなり, P は停止する．この操作を繰り返し, P を連結成分の中をすべて動かすと, その間に連結成分に含まれる Voronoi 点で停止する．停止点の数は $O(m)$ であることが証明できる．

補題 4.7. θ を固定したとき, すべての b-edge とそれに隣接する Voronoi 辺を取り除き, 残った Voronoi 辺を連結成分に分解する．このとき, 各連結成分においてどの b-edge の端点となっていない Voronoi 頂点の数は, $O(m)$ である．

次に, θ を 0 から 2π まで動かして, b-edge の端点でない v-point の数を評価する．まず, $\theta = 0$ の $P(0)$ -Euclid Voronoi 図からスタートして, Voronoi 図の位相が変わらない間の b-edge に無関係な v-point の数を数える．連結成分に接続する b-edge が変化しない間, 連結成分内の v-point は, 高々 $O(m^4)$ であり, 連結成分の数は, 高々 $O(mn)$ である．従って, b-edge に無関係な v-point の数は, $O(m^5 n)$ となる．次に, b-edge の変化が, 連結成分内の v-point の数にどのように影響するかについて調べる．ある b-edge が b-edge でなくなると, それに隣りあった Voronoi 辺で接続されている連結成分において, 新たな v-point が現われる可能性がある．それは, その b-edge と連結成分内の $O(m)$ 個の Voronoi 辺のうち 3 つの辺によって, v-point ができる場合である．このような v-point の数は, 各 b-edge に対して, $O(m^3)$ 個ある．他に, 配置可能領域の位相の変化なども考慮すると次の補題を得る．

補題 4.8. b-edge の端点でない v-point の数は, $O(m^4 n \lambda_{16}(mn))$ となる．

従って, θ ($0 \leq \theta \leq 2\pi$) が変化するときの $\tilde{P}(\theta)$ -Euclid Voronoi 図の組合せ的複雑度は, $O(m^4 n \lambda_{16}(mn))$ であることがわかる．

配置可能領域の位相の変化をすべて求めた後、すべての Voronoi 辺を扱うデータ構造としてヒープを用いて、平面走査法を使って問題 (P2) を解く。アルゴリズムについて、ここでは詳しいことは省くが、そのおもな手法を述べると次のようになる。各 Voronoi 辺に対して、 θ を 0 から 2π まで変化させたときその Voronoi 辺が退化する θ (v-point における θ の値) を求める。その θ の順で、すべての Voronoi 辺をヒープで管理する。退化する時の θ の値とそれに隣接する Voronoi 辺のデータを持っているものとする。アルゴリズムの各段階で、 θ を増加させたとき、どの Voronoi 辺が次に退化するかを求めておく。Voronoi 辺が退化するときには、新しく現われた辺とそれと隣接している Voronoi 辺をヒープにおいて更新する。Voronoi 領域を更新するとき、各 Voronoi 領域に対して、その Voronoi 領域が存在している θ の範囲を求め、与えられた範囲において、 P の任意の点と Q の任意の点との距離の最小値を計算する。これは、定数時間でできる。ここで、その最小値の最大値を覚えておくことにより、すべてのステップが終了した時に、問題 (P2) が解けたことになる。Voronoi 辺の総数は、 $O(mn)$ であるから、ヒープを更新する手間は、 $O(\log mn)$ である。また、 $\tilde{P}(\theta)$ -Euclid Voronoi 図の組合せ的複雑度は、 $O(m^4 n \lambda_{16}(mn))$ であることから、更新する回数も、 $O(m^4 n \lambda_{16}(mn))$ である。もう少し議論を続けると、次の定理を得る。

定理 4.1. $\tilde{P}(\theta)$ -Euclid Voronoi 図 ($0 \leq \theta \leq 2\pi$) の組合せ的複雑度は、 $O(m^4 n \lambda_{16}(mn))$ であり、問題 (P2) は $O(m^4 n \lambda_{16}(mn) \log mn)$ の手間で解くことができる。

5. 問題 (P3), (P4) について

パラメタ h を固定した場合、問題 (P3) は次のようにして解く。 k 個の Q を h ずつ平行にずらして置き、それらの共通部分を求める。そして、その共通部分に対して問題 (P1) を解けば求める解が得られる。この共通部分のサイズは、最悪の場合 $O(k^2 n^2)$ となるので、固定された h に対する (P3) は $O(mk^2 n^2 \log mkn)$ の手間で解くことができる。

h が定数ではないときは、 Q の k 個のコピーがパラメタ h で動いている場合の動的 P-Euclid Voronoi 図について調べなければならない。まず、この種の最も基本的な問題と考えることができる問題 (P4) を解く。つまり、 n 点からなる k 個の集合が、 n によらない低次の多項式または連続な有理関数にしたがって動くとき、その点集合に対する動的 Voronoi 図の組合せ的複雑度を計算する。すると、次の定理を得る。

定理 5.1. n 点からなる k 個の集合が各集合の形を変えずに動くとき、その動的 Voronoi 図の組合せ的複雑度は、 $k = 2, 3$ の時 $O(n^2)$ 、 $k \geq 4$ の時 $O(k^4 n^3 \log^* k)$ である。

これと同様のアプローチによって、問題 (P3) を解くと、次の定理が証明できる。

定理 5.2. 問題 (P3) は、 $k = 2$ の時 $O(m^2 n^2 \log mn)$ の手間、 $k \geq 3$ の時 $O(k^6 m^3 n^3 \log kmn)$ の手間で解くことができる。

6. おわりに

本稿では，凸多角形の多角形領域内への maximin 配置問題における何種類かの問題の解法と，それに関連した動的 Voronoi 図の組合せ的複雑度について述べた。

凸多角形の多角形領域内への maximin 配置問題は，いろいろな方面から研究されている興味深い問題である。本文中でも述べたように，地図に文字を配置するような場合にもあらわれる。この多角形配置問題を解くためには，動的 Voronoi 図についての研究が不可欠であった。凸多角形 P と多角形領域 Q に対して，新しい Voronoi 図である P-Euclid Voronoi 図を定義し，それを使って問題 (P1)，(P2)，(P3) を解いた。また，問題 (P3) に関しては，動いていく点に対する Voronoi 図を追跡していくという問題 (P4) の解法を用いた。

動的 Voronoi 図に関する研究は最近始められたばかりで，解かなければならない問題が多く残ってる。また，今回見てきたように，多角形の配置問題というような実際的な問題に深く関わっている。そういった意味でも，動的 Voronoi 図に関する研究は興味がある。

参考文献

- [AIK] H. Aonuma, H. Imai and Y. Kambayashi: A visual system of placing characters appropriately in multimedia map databases. In “*Visual Database Systems*” (Proc. the IFIP TC2/WG2.6 Working Conf. on Visual Database Systems), North-Holland, 1989, pp.525-546.
- [CK] L. P. Chew and K. Kedem: Placing the largest similar copy of a convex polygon among polygonal obstacles. *Proc. 5th ACM Symp. on Computational Geometry*, 1989, pp.167-174.
- [F1] S. Fortune: Fast algorithms for polygon containment. *Proc. 12th ICALP*, Lecture Notes in Computer Science 194, Springer-Verlag, 1985, pp.189-198.
- [F2] S. Fortune: A sweepline algorithm for Voronoi diagrams. *Algorithmica*, 2(1987), pp.153-174.
- [I] K. Imai: Voronoi diagrams for moving points (in Japanese). *Memories of Mathematical Research Institute*, Vol.695, Kyoto University, 1989, pp.225-275.
- [ISI] K. Imai, S. Sumino and H. Imai: Geometric fitting of two corresponding sets of points. *Proc. 5th ACM Symp. on Computational Geometry*, 1989, pp.266-275.